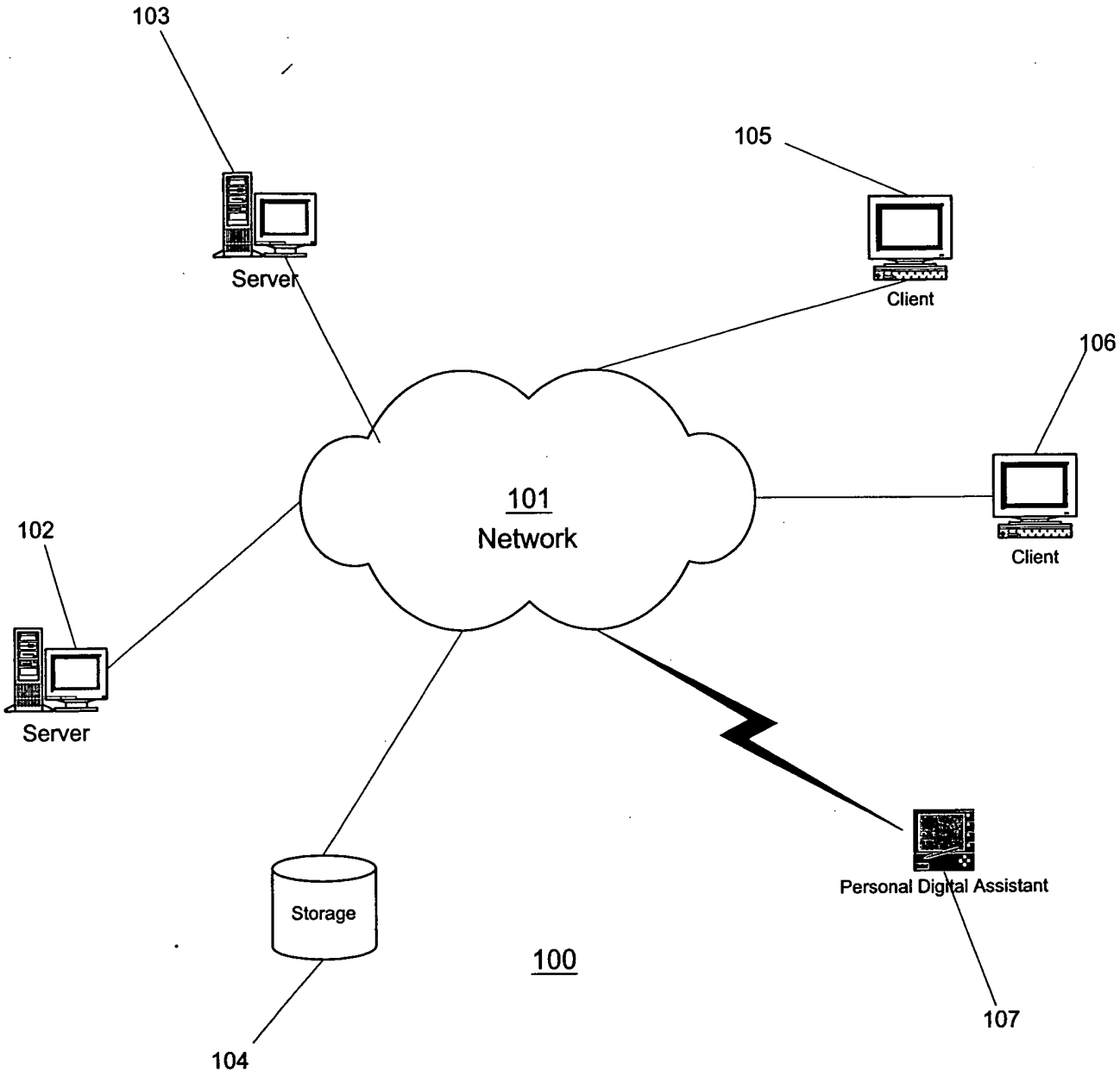
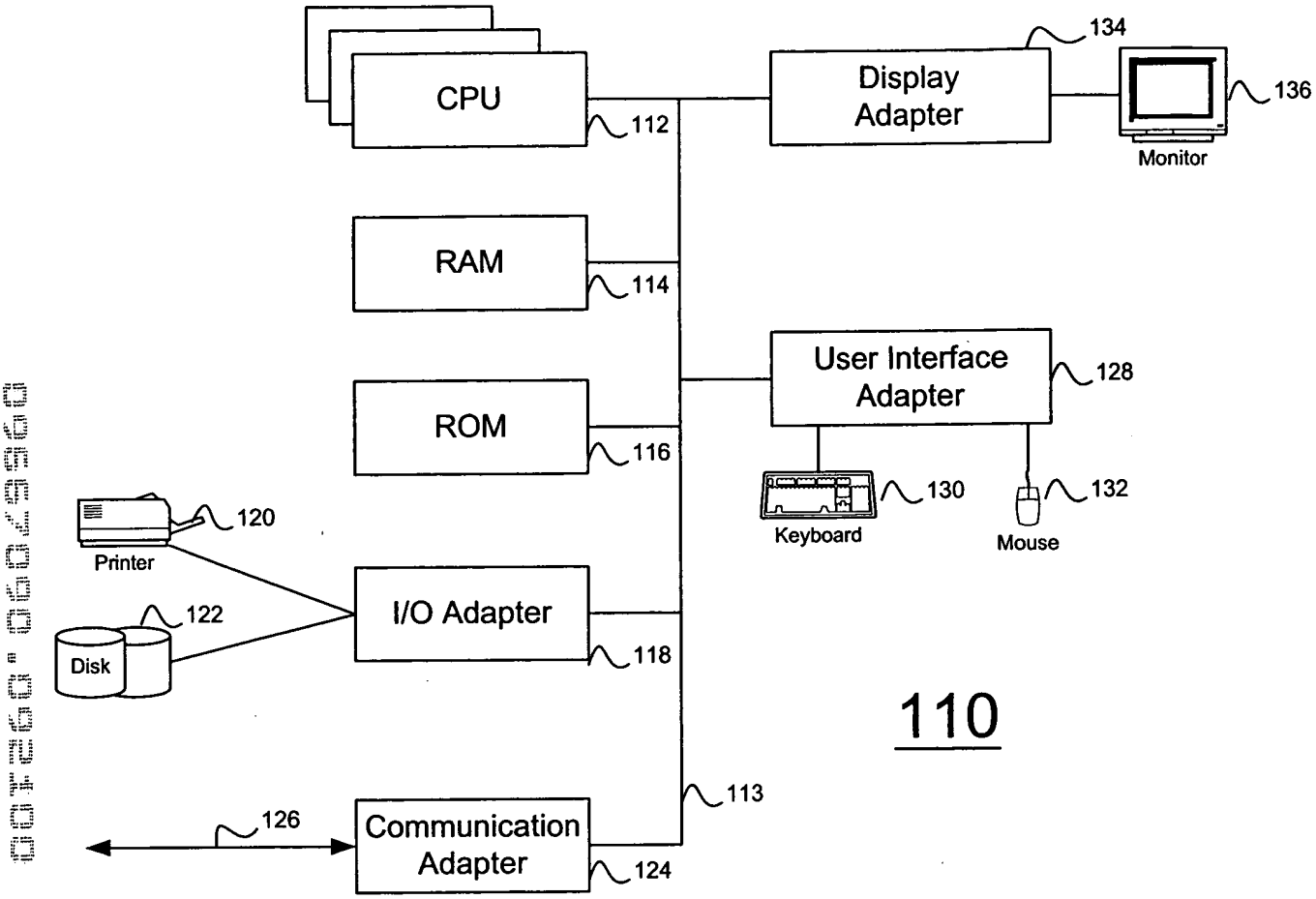


Prior Art

Figure 1A



Prior Art

Figure 1B

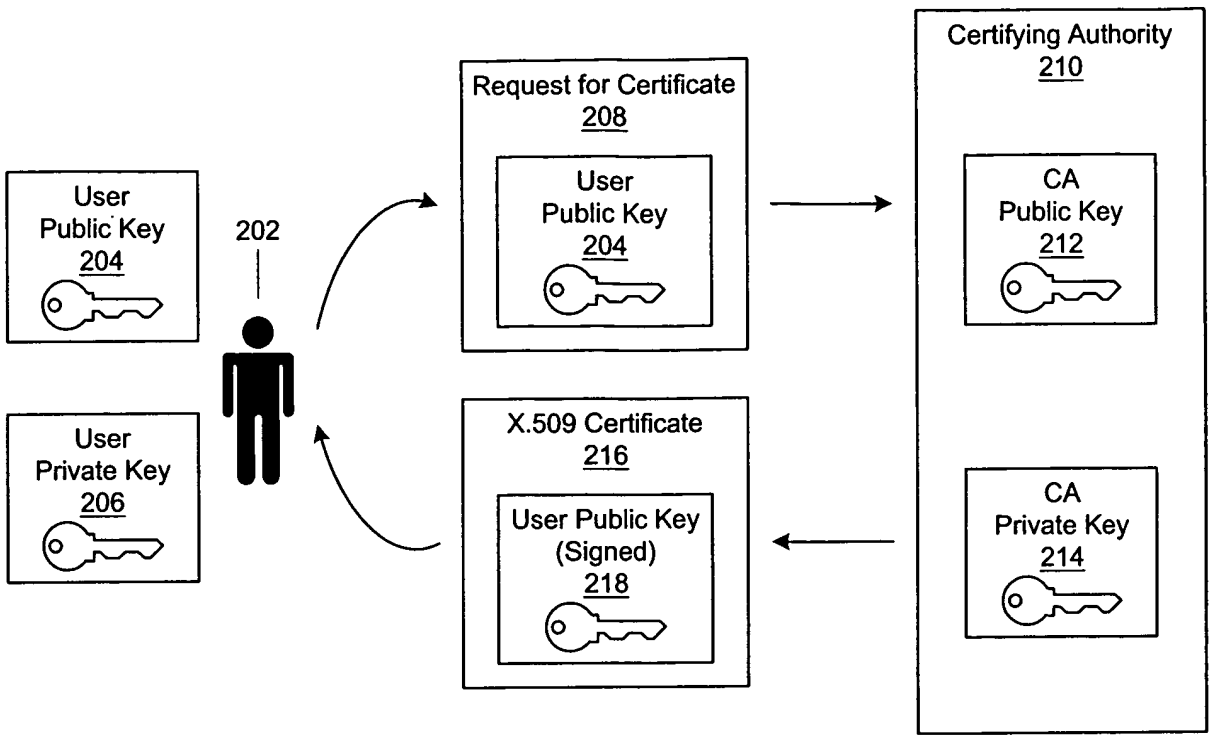


Figure 2

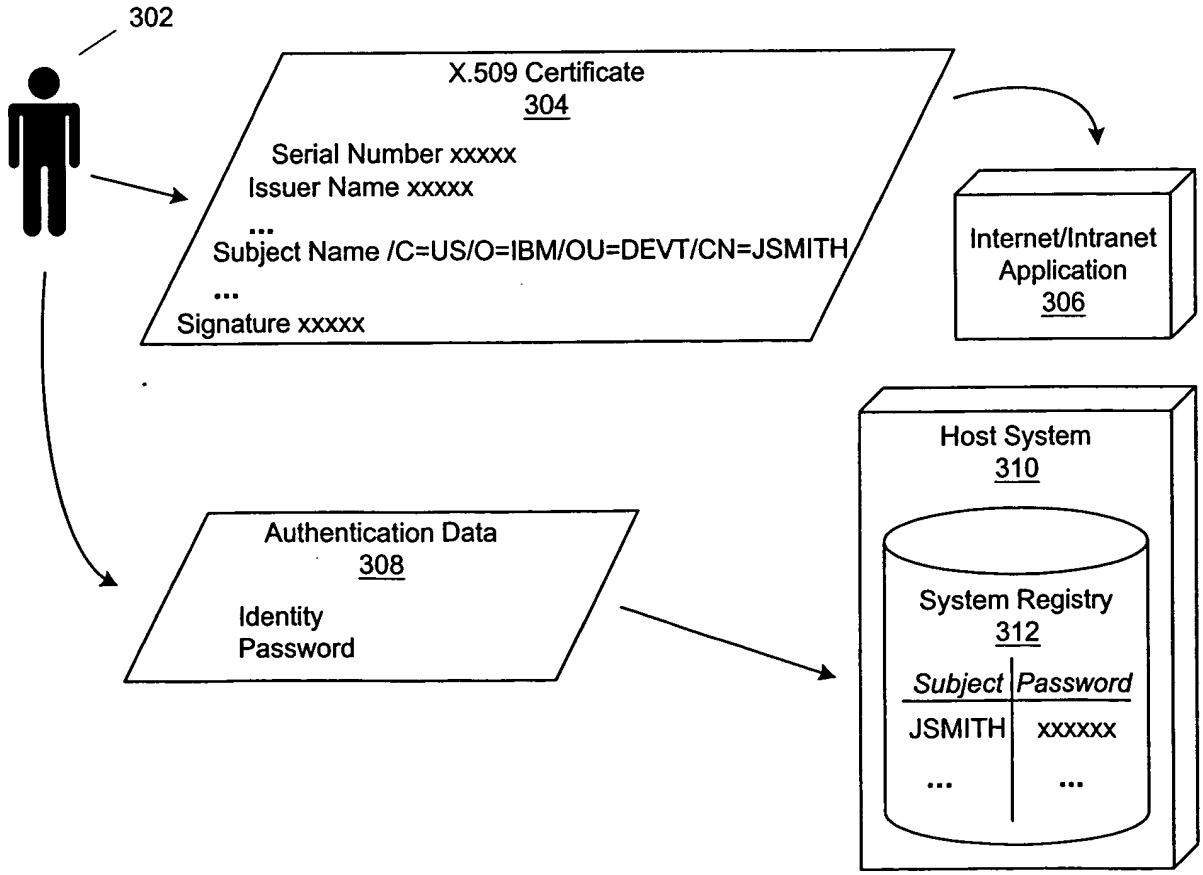


Figure 3

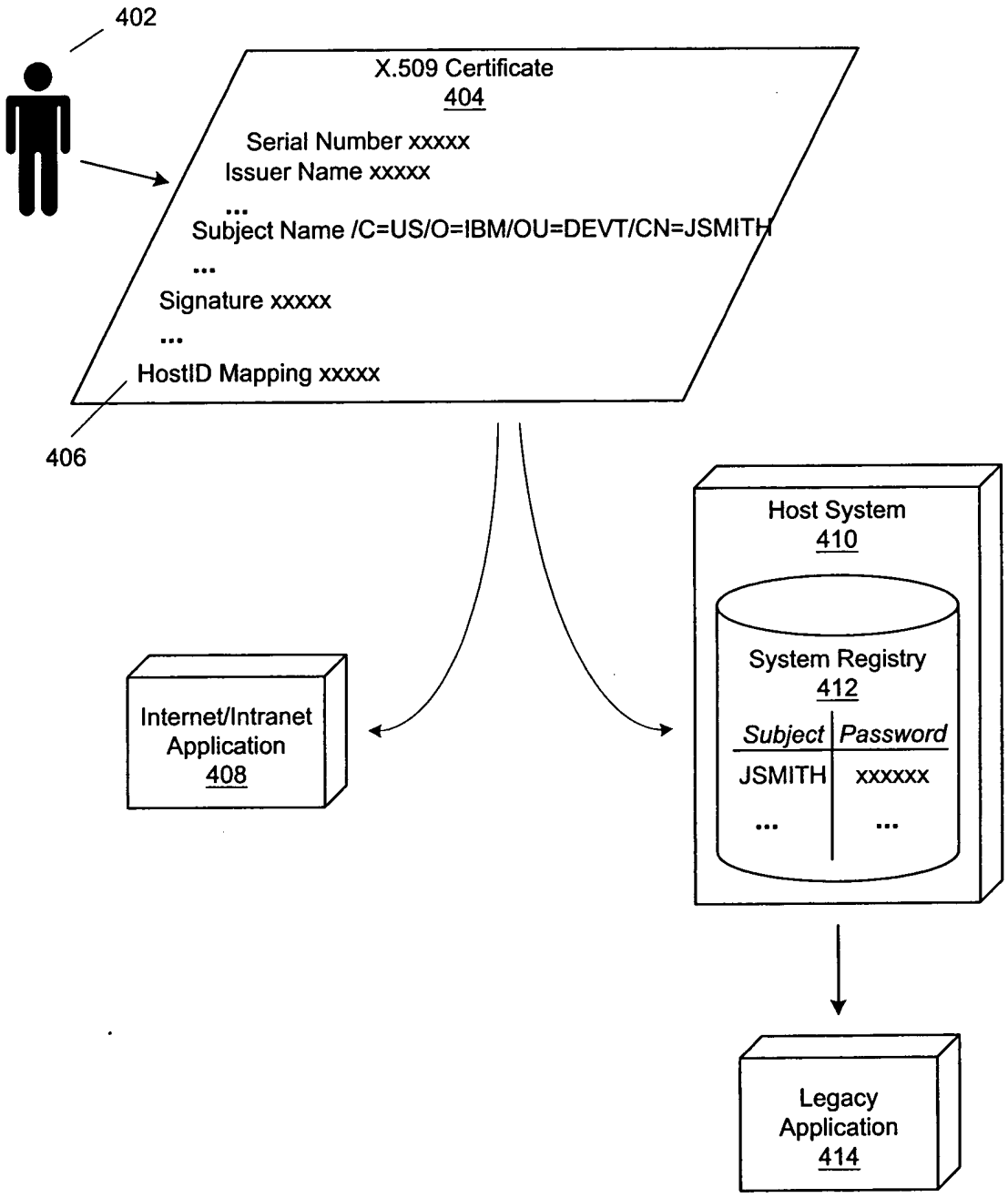

402
X.509 Certificate
404
Serial Number xxxxx
Issuer Name xxxxx
...
Subject Name /C=US/O=IBM/OU=DEVT/CN=JSMITH
...
Signature xxxxx
...
HostID Mapping xxxxx
406
Host System
410
System Registry
412
Subject
Password
JSMITH
xxxxxx
...
...
Internet/Intranet
Application
408
Legacy
Application
414

Figure 4

```
Certificate  ::=  SEQUENCE  {
      tbsCertificate           TBSCertificate,
      signatureAlgorithm       AlgorithmIdentifier,
      signature                BIT STRING  }

TBSCertificate  ::=  SEQUENCE  {
      version            [0]   Version DEFAULT v1,
      serialNumber             CertificateSerialNumber,
      signature                AlgorithmIdentifier,
      issuer                   Name,
      validity                 Validity,
      subject                  Name,
      subjectPublicKeyInfo     SubjectPublicKeyInfo,
      issuerUniqueID     [1]   IMPLICIT UniqueIdentifier OPTIONAL,
      subjectUniqueID    [2]   IMPLICIT UniqueIdentifier OPTIONAL,
      extensions         [3]   Extensions OPTIONAL   }

Version  ::=  INTEGER  {  v1(0), v2(1), v3(2)  }

CertificateSerialNumber  ::=  INTEGER

Validity ::= SEQUENCE {
      notBefore                Time,
      notAfter                 Time }

Time ::= CHOICE {
      utcTime                  UTCTime,
      generalTime              GeneralizedTime }

UniqueIdentifier  ::=  BIT STRING

SubjectPublicKeyInfo  ::=  SEQUENCE  {
      algorithm                AlgorithmIdentifier,
      subjectPublicKey         BIT STRING  }

Extensions  ::=  SEQUENCE SIZE (1..MAX) OF Extension

Extension  ::=  SEQUENCE  {
      extnID                   OBJECT IDENTIFIER,
      critical                 BOOLEAN DEFAULT FALSE,
      extnValue                OCTET STRING  }
```

Figure 5

Priort Art

```
HostIdMapping  ::=  SEQUENCE  {
      hostName           [1]   IMPLICIT IA5String,
      subjectID                IMPLICIT IA5String,
      proofOfIdPossession      IdProof OPTIONAL  }

IdProof  ::=  SEQUENCE  {
      secret                   OCTET STRING,
      encryptionAlgorithm      OBJECT IDENTIFIER  }
```

Figure 6

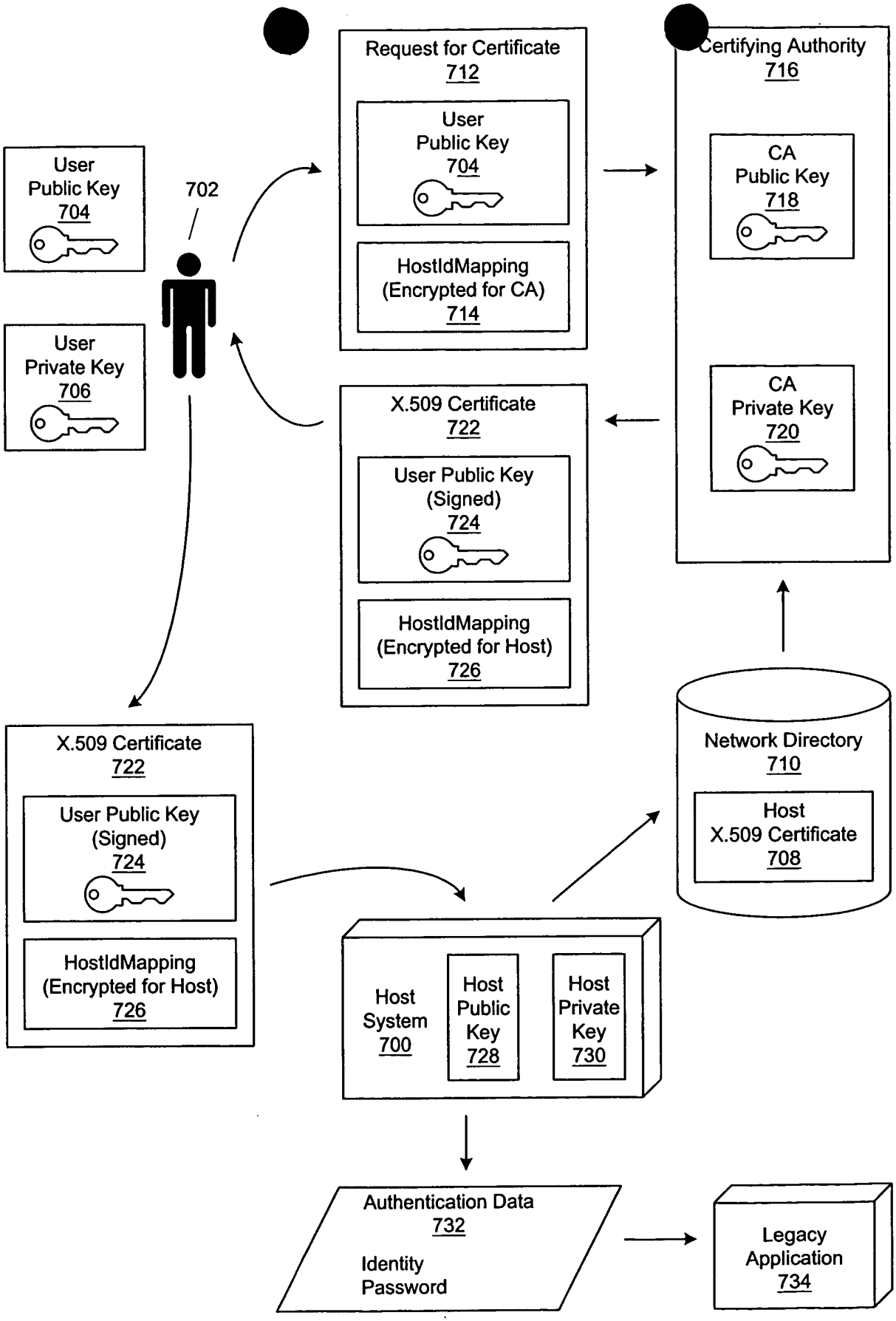

Request for Certificate
712
User
Public Key
704
HostIdMapping
(Encrypted for CA)
714
Certifying Authority
716
CA
Public Key
718
CA
Private Key
720
User
Public Key
704
702
User
Private Key
706
X.509 Certificate
722
User Public Key
(Signed)
724
HostIdMapping
(Encrypted for Host)
726
Network Directory
710
Host
X.509 Certificate
708
X.509 Certificate
722
User Public Key
(Signed)
724
HostIdMapping
(Encrypted for Host)
726
Host
System
700
Host
Public
Key
728
Host
Private
Key
730
Authentication Data
732
Identity
Password
Legacy
Application
734

Figure 7

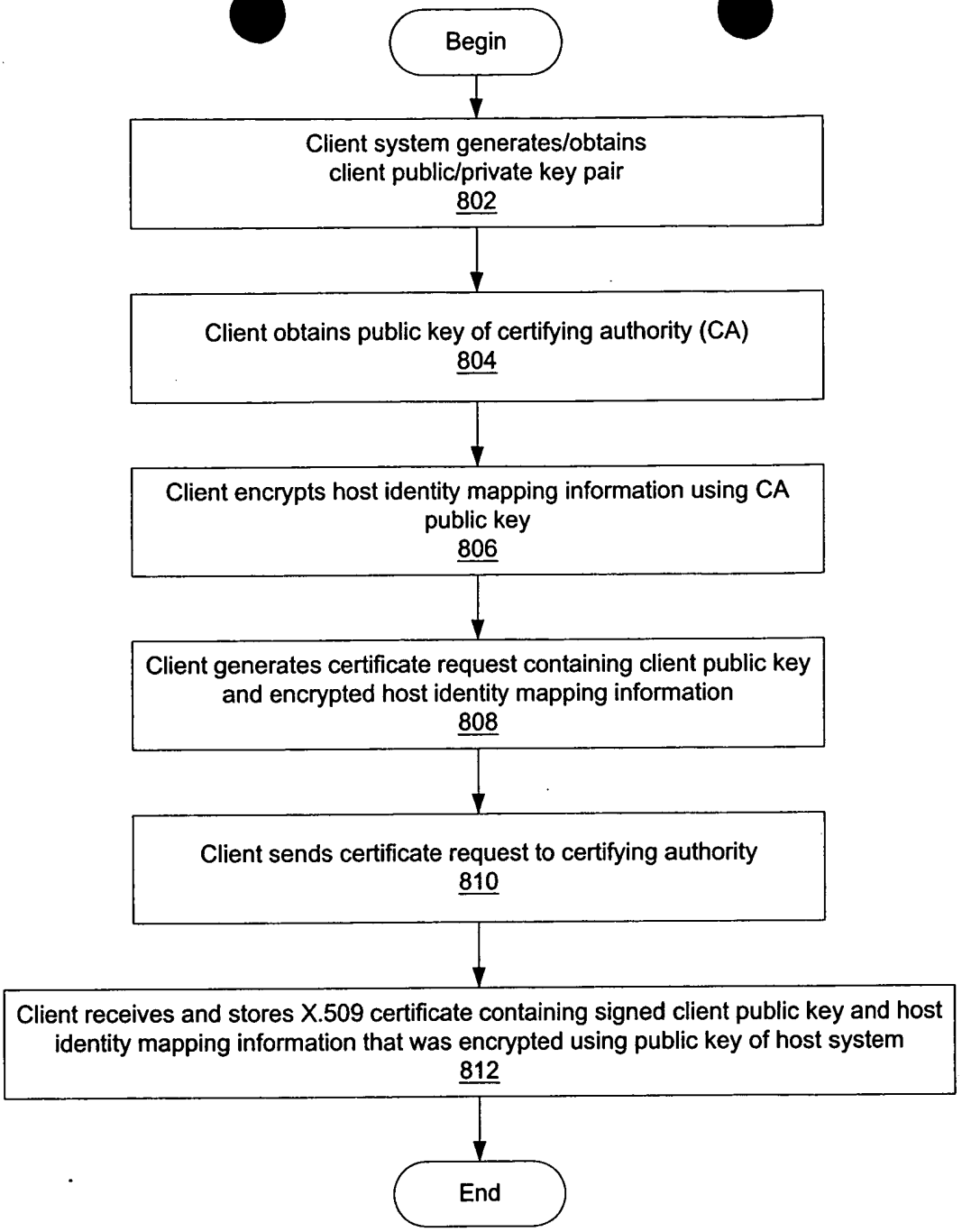

Begin
Client system generates/obtains client public/private key pair
802
Client obtains public key of certifying authority (CA)
804
Client encrypts host identity mapping information using CA public key
806
Client generates certificate request containing client public key and encrypted host identity mapping information
808
Client sends certificate request to certifying authority
810
Client receives and stores X.509 certificate containing signed client public key and host identity mapping information that was encrypted using public key of host system
812
End

Figure 8A

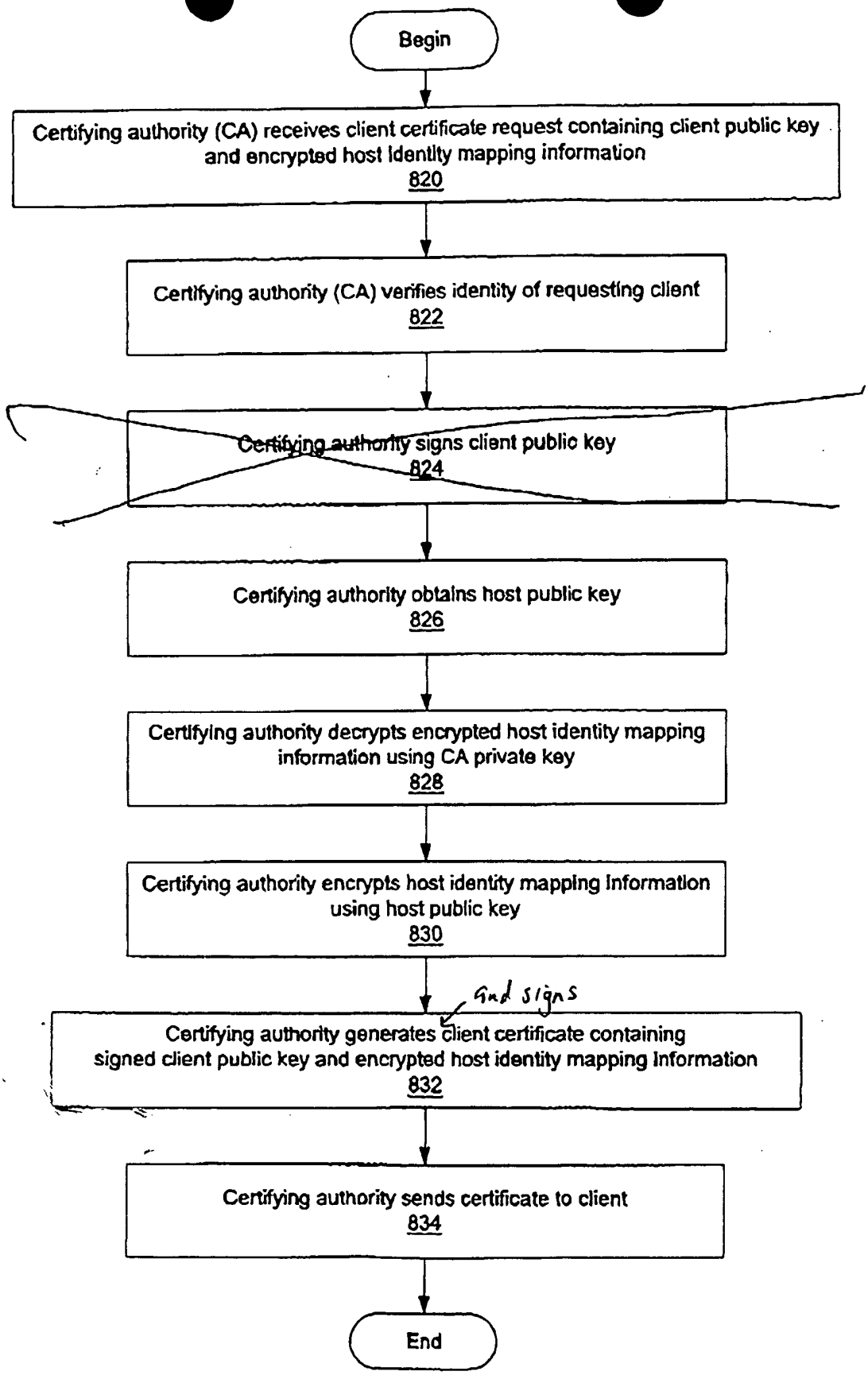

Begin
Certifying authority (CA) receives client certificate request containing client public key and encrypted host identity mapping information
820
Certifying authority (CA) verifies identity of requesting client
822
Certifying authority signs client public key
824
Certifying authority obtains host public key
826
Certifying authority decrypts encrypted host identity mapping information using CA private key
828
Certifying authority encrypts host identity mapping information using host public key
830
and signs
Certifying authority generates client certificate containing signed client public key and encrypted host identity mapping information
832
Certifying authority sends certificate to client
834
End

Figure 8B

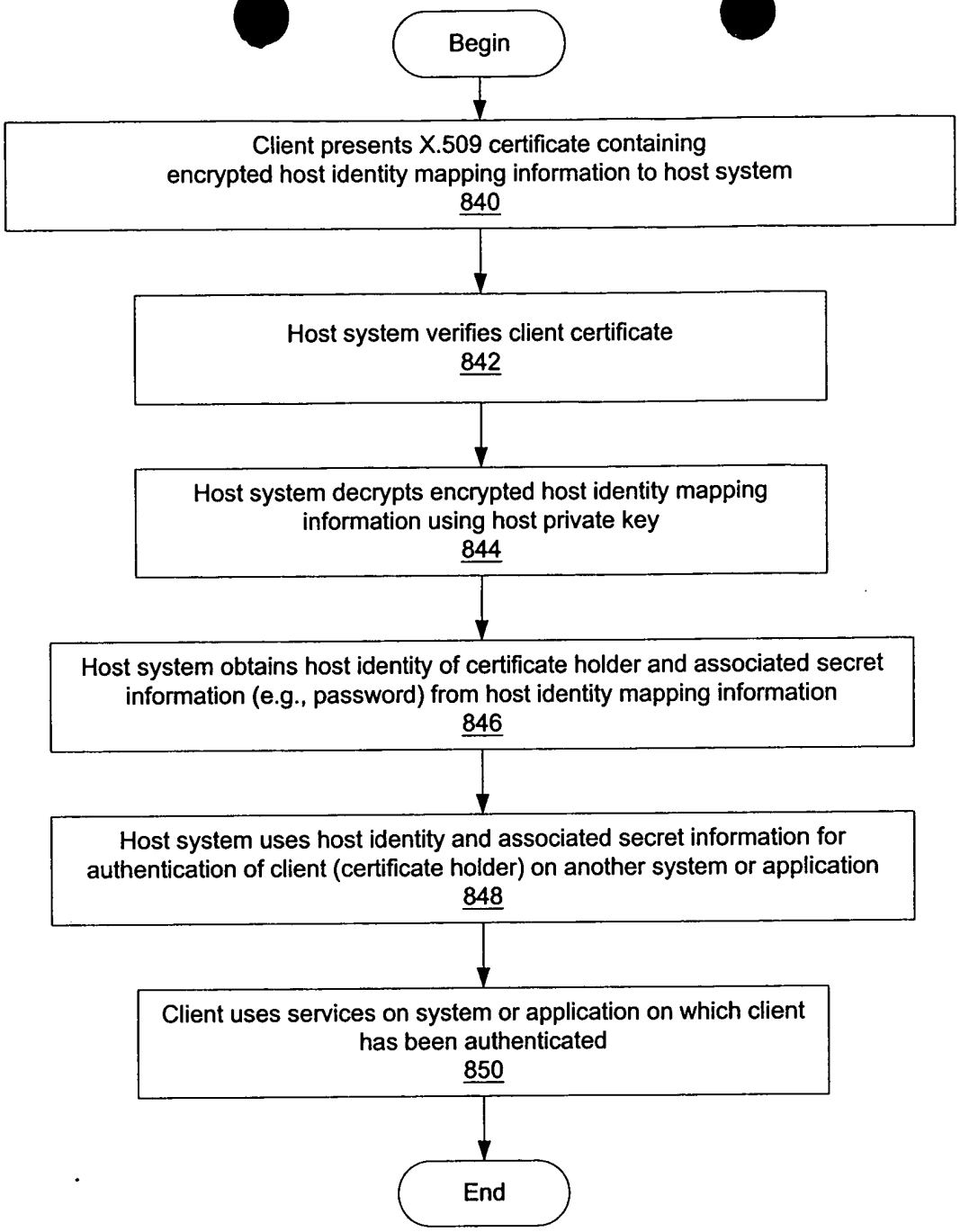

Begin
Client presents X.509 certificate containing encrypted host identity mapping information to host system
840
Host system verifies client certificate
842
Host system decrypts encrypted host identity mapping information using host private key
844
Host system obtains host identity of certificate holder and associated secret information (e.g., password) from host identity mapping information
846
Host system uses host identity and associated secret information for authentication of client (certificate holder) on another system or application
848
Client uses services on system or application on which client has been authenticated
850
End

Figure 8C